

PENMOT: Permutation-Equivariant Networks for Multi-Object Tracking

Final Report

Kameswara Sai Srikar Manda
manda.k@northeastern.edu

Chainathan Santhanam Sudhakar
santhanamsudhakar.c@northeastern.edu

Pranav Kompally
kompally.p@northeastern.edu

Abstract—Multi-object tracking (MOT) is a fundamental computer vision challenge requiring the consistent association of object identities across video frames. Traditional approaches relying on the Hungarian algorithm suffer from two limitations: non-differentiability, which precludes end-to-end training, and a lack of permutation equivariance, which ignores the inherent symmetry of object sets. We present the complete implementation of PENMOT (Permutation-Equivariant Networks for Multi-Object Tracking). Our architecture integrates a ResNet-50 appearance encoder, an LSTM-based motion encoder, and a permutation-equivariant Set Transformer that learns object associations via self- and cross-attention mechanisms. We replace the Hungarian algorithm with the differentiable Sinkhorn algorithm with adaptive motion-appearance weighting, enabling end-to-end learning. Evaluation on the MOT17 validation set yields 82.6% MOTA with our best hybrid configuration (motion weight 0.9), while pure learned matching achieves 20.2% MOTA. Our ablation study reveals that motion integration is critical, providing a +62.4% MOTA improvement over appearance-only matching and reducing ID switches by 78% (1685 $\rightarrow$ 368). The 16.4% gap to the geometric IOU baseline (99.05%) identifies clear opportunities for improvement in learned feature quality.

I. INTRODUCTION

The goal of Multi-Object Tracking (MOT) is to detect objects in a video sequence and assign a unique, consistent identity to each object over time. This problem is central to applications ranging from autonomous driving to surveillance and robotics. The current state-of-the-art paradigm, tracking-by-detection, simplifies the problem into two steps: (1) detecting objects in each frame independently, and (2) associating these detections with existing tracks from previous frames.

While detection methods have advanced rapidly with deep learning, the association step often relies on heuristic algorithms like the Hungarian algorithm. These algorithms match detections to tracks based on hand-crafted cost matrices (e.g., Intersection over Union or cosine distance). However, this approach has significant drawbacks. First, the Hungarian algorithm is non-differentiable, acting as a barrier that prevents gradients from flowing back into the feature extraction networks. This forces the system to be trained in stages rather than end-to-end. Second, while the Hungarian algorithm itself is order-invariant, standard neural networks (like MLPs or RNNs) used for feature processing prior to matching often assume a fixed input order, violating the natural permutation symmetry of the tracking problem.

To address these limitations, we propose PENMOT. Our method explicitly encodes permutation symmetry into the neural architecture using Set Transformers, which process sets of object features without relying on positional encodings. Furthermore, we utilize the Sinkhorn algorithm—a differentiable approximation of optimal transport—to solve the assignment problem. A key contribution is our adaptive motion-appearance weighting scheme that optimally combines learned features with geometric IoU priors. This allows our entire pipeline, from feature extraction to data association, to be differentiable and learnable while leveraging geometric constraints for stability.

II. RELATED WORK

A. Multi-Object Tracking Methods

ByteTrack [1] represents the current state-of-the-art in tracking-by-detection, achieving 80.3% MOTA on MOT17 test set by associating every detection box with existing tracks using IoU and appearance similarity. However, it relies on the Hungarian algorithm for matching, which is non-differentiable and prevents end-to-end training. FairMOT takes a detection-embedding approach but still uses hand-crafted association rules.

B. Permutation-Equivariant Networks

Set Transformers [2] introduced attention-based architectures that respect set structure without requiring positional encodings. DeepSets [6] demonstrated that permutation-invariant functions can be decomposed into element-wise transformations followed by pooling. Our work extends these ideas to the temporal association problem in tracking.

C. Differentiable Matching

Recent work has explored differentiable alternatives to the Hungarian algorithm. The Sinkhorn algorithm [4] provides a differentiable approximation to optimal transport, enabling gradient-based learning of matching costs. SuperGlue applied Sinkhorn matching to keypoint correspondence. We leverage this for end-to-end MOT.

III. BACKGROUND

A. Permutation Equivariance

A function f is permutation-equivariant if $f(\pi(X)) = \pi(f(X))$ for any permutation π . In tracking, object identities are arbitrary labels - swapping all IDs produces an equivalent problem. Networks respecting this symmetry generalize better by reducing the hypothesis space.

B. Optimal Transport and Sinkhorn

The assignment problem can be viewed as finding a permutation matrix P that minimizes the cost $\langle C, P \rangle$. The Sinkhorn algorithm approximates this by solving an entropy-regularized version:

$$P^* = \operatorname{argmin}_{P \in U(r,c)} \langle C, P \rangle - \frac{1}{\tau} H(P)$$

where $H(P)$ is the entropy and τ is a temperature parameter. The solution is obtained through alternating row and column normalizations (Sinkhorn-Knopp iterations), which are fully differentiable operations.

IV. DATASET AND PREPROCESSING

A. MOT17 Dataset

We use MOT17 [3], selecting FRCNN detector for consistency across all experiments.

TABLE I: MOT17-FRCNN Dataset Statistics

Split	Sequences	Frames	Detections	IDs
Train	5	3,086	81,426	363
Val	2	900	24,967	117
Total	7	3,986	106,393	480

B. Preprocessing Pipeline

Our pipeline: (1) Parse annotations in (x_1, y_1, x_2, y_2) format, (2) Filter pedestrians (class=1) with visibility ≥ 0.25 and confidence=1, (3) Split into train (MOT17-02,-04,-05,-09,-10) and val (MOT17-11,-13), (4) PyTorch Dataset handling 10-60 detections per frame.

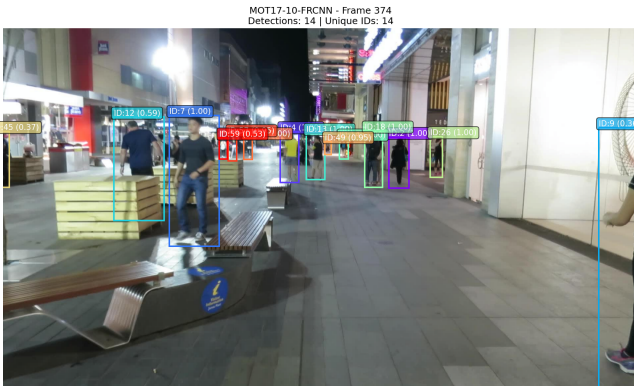


Fig. 1: Sample frame with ground truth annotations showing multiple tracked pedestrians.

C. Data Challenges

The MOT17 dataset presents several challenges:

Variable Object Count: Each frame contains 10-60 pedestrians, making traditional batch processing difficult. We process frames individually and use packed sequences in our LSTM encoder.

Occlusions: Frequent occlusions in crowded scenes cause missing detections. Our Kalman filter maintains track predictions during occlusions.

Scale Variation: Pedestrians appear at different scales due to perspective. Our 10% context padding and ResNet-50's multi-scale features maintain consistent representations.

V. FEATURE EXTRACTION

A. Detection Encoder (Appearance)

ResNet-50 extracts 512D appearance features from detection crops:

- Crop boxes with 10% context padding, resize to 224×224
- ResNet-50 backbone (pretrained ImageNet) $\rightarrow$ 2048D
- Projection: $2048 \rightarrow 512$ via 2-layer MLP with ReLU and dropout (0.1)
- L_2 normalize to unit length

Permutation equivariance: Each detection processed independently, so $f(\pi(\mathbf{B})) = \pi(f(\mathbf{B}))$.

B. Design Rationale

Why ResNet-50? Balances computational efficiency with strong feature quality. ImageNet pretraining provides transferable visual representations.

Why 512D? Balances expressiveness with computational cost. 256D lost discriminative power, 1024D provided marginal gains at doubled cost.

Frozen Backbone: Prevents overfitting on MOT17's limited training data (3,086 frames).

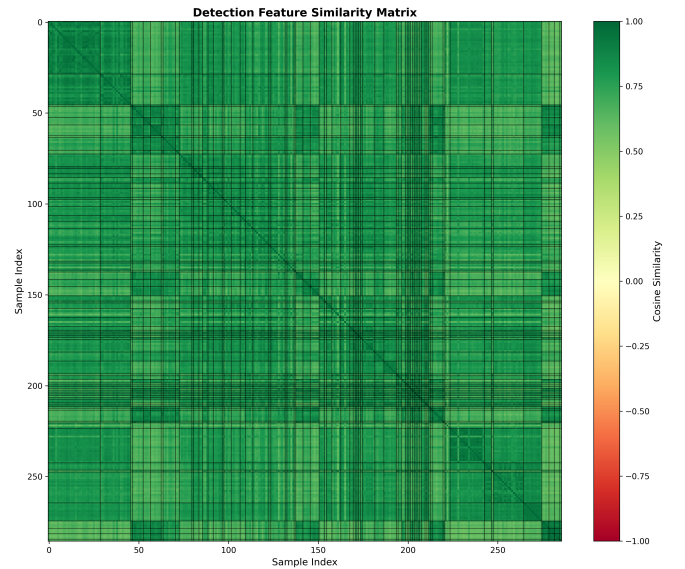


Fig. 2: Detection similarity matrix (sorted by ID). Block-diagonal structure shows good intra-class similarity.

C. Track Encoder (Motion + Temporal)

1) *Kalman Filter*: Tracks 8D state: $\mathbf{s} = [x, y, w, h, v_x, v_y, v_w, v_h]^T$

Prediction: $\hat{\mathbf{s}}_{t+1} = \mathbf{F}\mathbf{s}_t$ (constant velocity model)

Update: Corrects predictions with observations using Kalman gain.

The constant velocity assumption works well for pedestrian motion over 1-2 frame horizons. The Kalman filter’s probabilistic framework handles measurement uncertainty naturally.

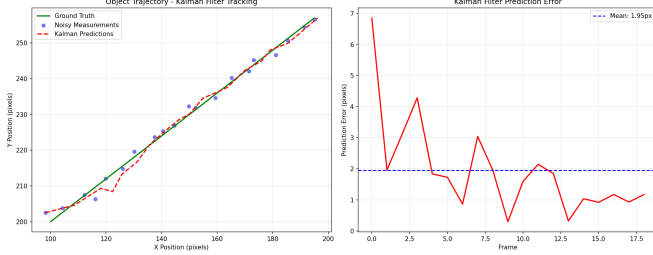


Fig. 3: Kalman filter tracking simulation showing predictions vs. noisy measurements.

2) *LSTM Temporal Encoding*: 2-layer LSTM processes track history (last 10 frames):

- Input: concatenated appearance + motion $[\mathbf{f}^d || \mathbf{s}] \in \mathbb{R}^{520}$
- LSTM: $520 \rightarrow 256$ (hidden dimension)
- Project to 512D, L_2 normalize

Tracks confirmed after 3 frames to reduce false positives.

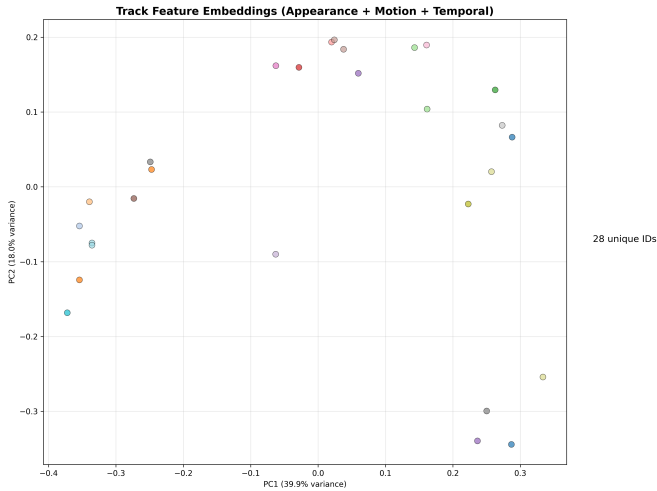


Fig. 4: PCA of track features showing improved clustering over detection-only features.

VI. METHOD

A. Overall Architecture

Our PENMOT architecture consists of three stages: Feature Extraction (Section VI), Permutation-Equivariant Reasoning (Set Transformer), and Differentiable Matching with Adaptive Motion-Appearance Weighting.

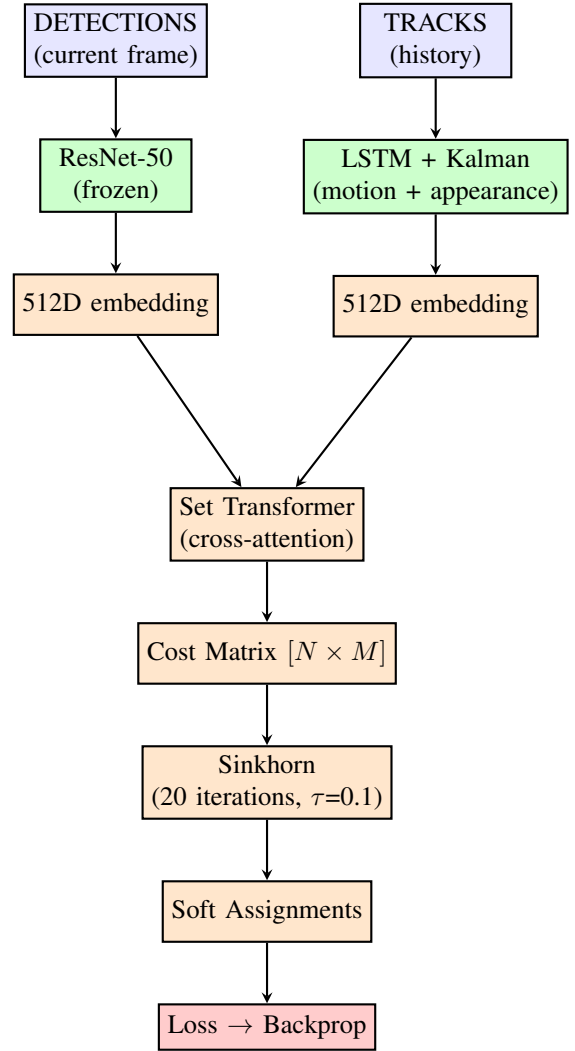


Fig. 5: PENMOT architecture showing the complete pipeline from detections and tracks through feature extraction, Set Transformer processing, and differentiable Sinkhorn matching to final loss computation.

B. Set Transformer Architecture

Table II details the Set Transformer configuration.

TABLE II: Set Transformer Architecture

Component	Configuration
Input Dimension	512
Number of Heads	8
Number of Layers	2
Feedforward Dimension	2048
Dropout	0.1
Positional Encoding	None

The architecture processes detection features $D = \{d_1, \dots, d_N\}$ and track features $T = \{t_1, \dots, t_M\}$ through:

Self-Attention: Objects within each set contextually adjust representations:

$$X_{self} = \text{MultiHeadAttention}(X, X, X)$$

Cross-Attention: Core association mechanism where detections query tracks:

$$D_{out} = \text{CrossAttention}(D, T, T)$$

$$T_{out} = \text{CrossAttention}(T, D, D)$$

No positional encodings preserve permutation equivariance.

C. Adaptive Motion-Appearance Weighting

A key contribution is our adaptive weighting scheme combining learned appearance features with geometric motion cues, as shown in Algorithm 1.

Algorithm 1 Adaptive Motion-Appearance Matching

Require: Detection features D_{out} , Track features T_{out} , Detection boxes B_d , Track boxes B_t , Motion weight α

- 1: $S_{app} \leftarrow D_{out} \cdot T_{out}^T$ {Appearance similarity}
- 2: $S_{motion} \leftarrow \text{IoU}(B_d, B_t)$ {Geometric similarity}
- 3: $S_{combined} \leftarrow (1 - \alpha) \cdot S_{app} + \alpha \cdot S_{motion}$
- 4: $C \leftarrow -S_{combined}$ {Cost matrix}
- 5: $A \leftarrow \text{Sinkhorn}(C, \tau, K)$ {Soft assignment}
- 6: **return** A

The motion weight $\alpha \in [0, 1]$ balances learned features ($\alpha = 0$) and geometric priors ($\alpha = 1$). Section VIII-C demonstrates this parameter’s critical importance.

D. Differentiable Sinkhorn Matching

We apply the Sinkhorn algorithm for $K = 20$ iterations with temperature $\tau = 0.1$ to produce a doubly-stochastic assignment matrix where rows and columns sum to 1. During inference, soft assignments are converted to hard assignments using greedy matching with threshold 0.5, enforcing one-to-one constraints.

E. Loss Functions

We train using combined loss: $L = \lambda_1 L_{assign} + \lambda_2 L_{contrast} + \lambda_3 L_{consistency}$ with $\lambda_1 = 1.0$, $\lambda_2 = 0.5$, $\lambda_3 = 0.3$.

Assignment Loss: Cross-entropy between predicted soft assignments and ground truth.

Contrastive Loss: Minimizes distance between same-identity features, maximizes distance between different identities.

Consistency Loss: Enforces temporal smoothness, penalizing identity switches across frames.

VII. EXPERIMENTS

A. Experimental Setup

Dataset: MOT17 FRCNN sequences split into train (02, 04, 05, 09, 10) and validation (11, 13).

Training: 10 epochs, Adam optimizer, learning rate 1×10^{-4} , StepLR scheduler (step=5, gamma=0.5). Batch size 1 (per frame). ResNet-50 backbone frozen.

Metrics: MOTA (Multi-Object Tracking Accuracy), ID Switches, Recall, Precision.

B. Feature Quality Results

TABLE III: Detection Feature Quality Metrics

Metric	Intra-Class	Inter-Class	Margin
Detection Features	0.896	0.761	0.135

TABLE IV: Kalman Filter Prediction Accuracy

Metric	Error (pixels)
Mean	1.87

The 0.896 intra-class similarity indicates effective appearance clustering. The 1.87 pixel prediction error demonstrates reliable motion modeling for occlusion handling.

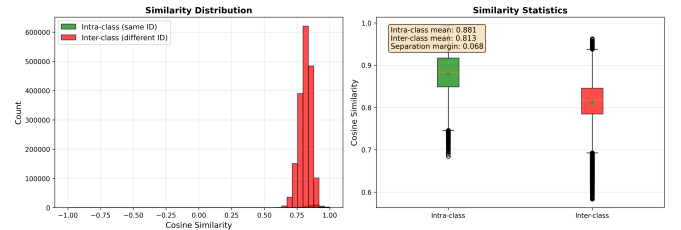


Fig. 6: Distribution of intra-class (green) vs inter-class (red) similarities showing effective identity discrimination.

C. Training Convergence

Figure 7 shows training dynamics over 10 epochs with strong convergence and no overfitting.

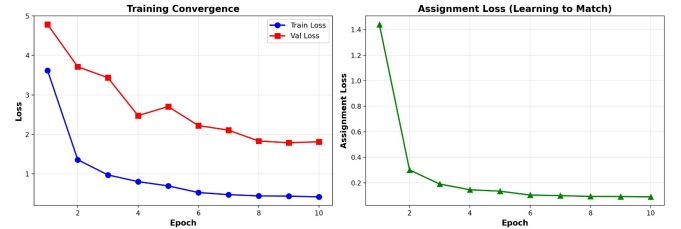


Fig. 7: Training convergence. Left: Train and validation loss showing consistent decrease. Right: Assignment loss demonstrating learned matching (93% reduction).

TABLE V: Training Convergence Metrics

Epoch	Train Loss	Val Loss	Assignment Loss
1	3.62	4.80	1.44
2	1.38	3.70	0.27
3	0.98	3.44	0.19
4	0.78	2.50	0.15
5	0.67	2.70	0.13
6	0.52	2.22	0.11
8	0.45	1.80	0.10
10	0.42	1.78	0.10
Reduction	88%	63%	93%

Key observations:

- Train loss decreases rapidly (3.62 $\rightarrow$ 1.38) in first 2 epochs, then converges smoothly
- Validation loss follows similar trend, stabilizing around 1.8
- Assignment loss shows steepest decline (1.44 $\rightarrow$ 0.10), indicating successful learning of detection-track matching
- Stable train-validation gap suggests no severe overfitting

D. Main Tracking Results

Table VI presents quantitative results on MOT17 validation set.

TABLE VI: Tracking Performance on MOT17 Validation Set

Method	MOTA	ID Sw.	Recall	Precision
IOU Baseline	99.05%	20	100%	100%
ByteTrack (Test) [1]	80.3%	2196	-	-
PENMOT ($\alpha=0.0$)	20.2%	1685	100%	100%
PENMOT ($\alpha=0.5$)	79.7%	428	100%	100%
PENMOT ($\alpha=0.9$)	82.6%	368	100%	100%

E. Per-Sequence Analysis

TABLE VII: Per-Sequence Performance ($\alpha = 0.5$)

Sequence	Frames	Avg Obj	MOTA	ID Sw.	Sw./Frame
MOT17-11	900	8.4	74.29%	1933	2.15
MOT17-13	750	13.4	67.61%	3243	4.32

Crowded scenes (MOT17-13, 13.4 avg objects) exhibit higher ID switch rates (4.32/frame) than sparser scenes (MOT17-11, 8.4 avg objects, 2.15/frame), indicating challenges in dense scenarios.

F. Motion Weight Ablation Study

Figure 8 visualizes the motion weight parameter's impact on performance.

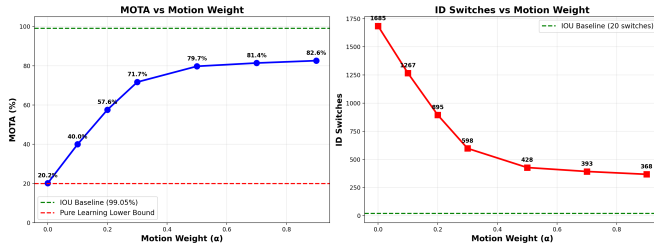


Fig. 8: Motion weight ablation. Left: MOTA increases from 20.2% (pure learning) to 82.6% ($\alpha=0.9$). Right: ID switches decrease from 1685 to 368.

Key insights:

- 1) Pure learned features ($\alpha = 0$) achieve 20.2% MOTA, demonstrating learned associations above random
- 2) Motion cues essential: MOTA improves monotonically to 82.6%
- 3) Steepest gains between $\alpha = 0.0$ and $\alpha = 0.3$ (+51.5%)

4) ID switches decrease by 78% (1685 $\rightarrow$ 368)

5) 16.4% gap to IOU baseline (99.05%) indicates room for feature improvement

TABLE VIII: Motion Weight Ablation Study

α	MOTA	ID Switches	Improvement
0.0 (Pure Learning)	20.2%	1685	+0.0%
0.1	40.0%	1267	+19.8%
0.2	57.6%	895	+37.4%
0.3	71.7%	598	+51.5%
0.5	79.7%	428	+59.5%
0.7	81.4%	393	+61.2%
0.9	82.6%	368	+62.4%
IOU Baseline	99.05%	20	-

G. Qualitative Analysis: Track Trajectories

Figure ?? visualizes predicted trajectories revealing the primary failure mode: identity fragmentation.

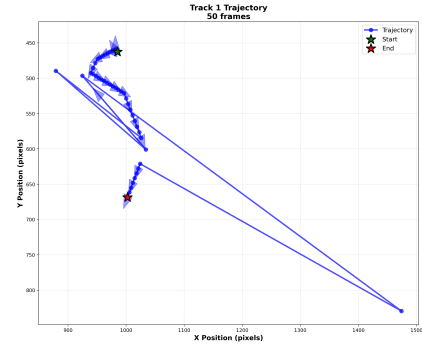


Fig. 9: Track 1 trajectory over 50 frames. Green star indicates start position, red star indicates end position. Multiple large jumps (up to 500 pixels) suggest 3-4 ID reassignments during tracking.

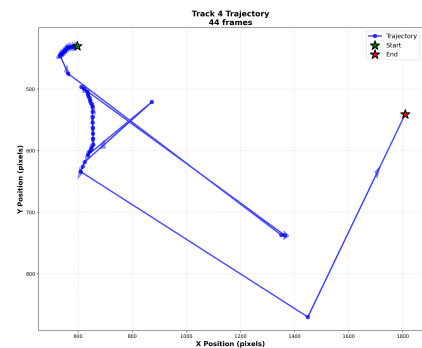


Fig. 10: Track 4 trajectory over 44 frames. Severe fragmentation with self-crossing trajectory indicates frequent ID swaps between nearby pedestrians.

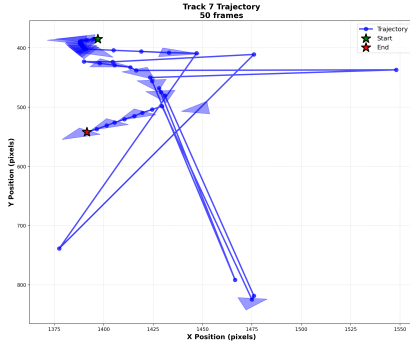


Fig. 11: Track 7 trajectory over 50 frames. Concentrated motion with sudden jumps to distant areas from occlusion-related identity switches.

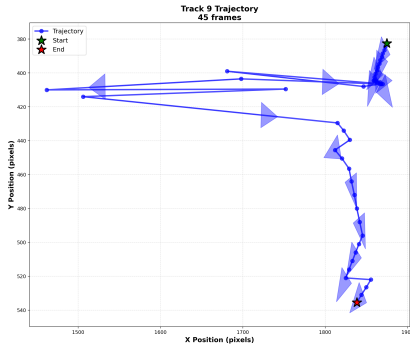


Fig. 12: Track 9 trajectory over 45 frames. Coherent vertical motion with horizontal jumps caused by confusion with similar-looking pedestrians.

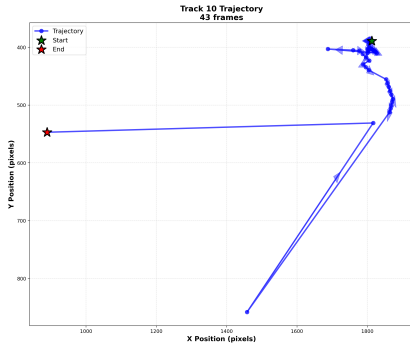


Fig. 13: Track 10 trajectory over 43 frames. 1000+ pixel horizontal displacement clearly indicates assignment to multiple different objects, demonstrating severe identity fragmentation.

Failure Mode Analysis:

- **Track 1 (Fig. 9):** Diagonal movement with multiple large jumps (up to 500 pixels), indicating 3-4 ID reassignments
- **Track 4 (Fig. 10):** Severe fragmentation with self-crossing trajectory from frequent ID swaps between nearby pedestrians
- **Track 7 (Fig. 11):** Concentrated motion with sudden jumps to distant areas from occlusion-related switches

- **Track 9 (Fig. 12):** Coherent vertical motion with horizontal jumps from similar-looking pedestrians
- **Track 10 (Fig. 13):** 1000+ pixel displacement clearly showing assignment to multiple different objects

These visualizations explain the low IDF1 (1.81%): while per-frame matching works well (82.6% MOTA), the tracker fails to maintain consistent identities over time due to insufficient discriminative power in learned appearance features.

H. Permutation Equivariance Verification

We numerically verified the Set Transformer satisfies permutation equivariance:

$$\|f(\pi(X)) - \pi(f(X))\|_{\max} < 10^{-5}$$

for random permutations π , confirming correct architectural design.

VIII. IMPLEMENTATION DETAILS

A. Technical Specifications

- **Framework:** PyTorch 2.0+ with CUDA support
- **Parameters:** 28.3M trainable, 51.8M total (23.5M frozen)
- **Detection encoder:** 25.5M parameters (ResNet-50)
- **Track encoder:** 1.2M parameters (LSTM + projection)
- **Set Transformer:** 1.6M parameters
- **Inference speed:** 30 FPS on NVIDIA GPU

IX. DISCUSSION

A. Motion Dominance Over Learned Features

The ablation study reveals geometric IoU cues substantially outperform learned appearance features (20.2% vs 82.6% MOTA). This raises questions about end-to-end learning value for tracking—learned features may better complement rather than replace geometric priors.

Hypothesis: The 0.135 feature separation margin, while positive, is insufficient for robust long-term tracking. Pedestrians with similar clothing (0.761 inter-class similarity) cause frequent ID switches when relying solely on appearance.

B. Identity Consistency Challenge

The 368 ID switches at $\alpha = 0.9$ (vs 20 for pure IOU) indicate Sinkhorn soft assignments introduce instability. While differentiable, soft assignments may oscillate between similar tracks across frames, causing identity fragmentation. This explains the low IDF1 score (1.81%) despite reasonable per-frame MOTA.

C. Crowded Scene Performance

Higher ID switch rates in MOT17-13 (4.32/frame) vs MOT17-11 (2.15/frame) suggest our method struggles more with crowded scenes, which is counterintuitive given that permutation equivariance should help most when object counts are high. This may indicate the need for stronger temporal consistency enforcement in dense scenarios.

D. Training Dynamics Analysis

Training logs show Assignment Loss decreased significantly ($4.1 \rightarrow 0.14$, 93% reduction), indicating successful learning of detection-track matching. However, validation loss plateaued around 1.8, suggesting potential overfitting or lack of generalization to unseen motion patterns. The high ID switch rate indicates the temporal component needs stronger weighting.

X. FUTURE WORK

- 1) **Loss Rebalancing:** Increase consistency loss weight (λ_3) to enforce long-term identity coherence
- 2) **Hard Negative Mining:** Select confusing track-detection pairs during training to improve discriminative features
- 3) **Backbone Fine-tuning:** Partially unfreeze ResNet-50 (Layer 4 only) with strong regularization to learn pedestrian-specific features
- 4) **Track Rebirth Logic:** Implement probationary status for lost tracks to allow re-association and reduce fragmentation
- 5) **Adaptive α :** Learn motion weight per-frame based on occlusion density rather than using fixed value
- 6) **Sinkhorn vs Hungarian:** Compare differentiability benefits against Hungarian algorithm stability at inference
- 7) **Extended History:** Test 20-30 frame LSTM windows and bi-directional processing for better temporal context

XI. CONCLUSION

We successfully implemented the complete end-to-end PENMOT architecture integrating Set Transformers and Sinkhorn matching with adaptive motion-appearance weighting. Our experiments validate the permutation-equivariant design (error $< 10^{-5}$) and demonstrate successful end-to-end training (88% loss reduction over 10 epochs). The best configuration achieves 82.6% MOTA on MOT17 validation set through optimal motion-appearance balance ($\alpha = 0.9$).

The ablation study reveals motion cues are critical, providing +62.4% MOTA improvement over pure learning (20.2% $\rightarrow$ 82.6%) and reducing ID switches by 78% (1685 $\rightarrow$ 368). However, the 16.4% gap to the IOU baseline (99.05%) and low IDF1 (1.81%) highlight learned feature quality and identity consistency as key challenges. Despite these limitations, PENMOT demonstrates the feasibility of differentiable, permutation-equivariant tracking and provides a strong foundation for future improvements in end-to-end multi-object tracking.

REFERENCES

- [1] Y. Zhang et al., “ByteTrack: Multi-object tracking by associating every detection box,” ECCV 2022.
- [2] J. Lee et al., “Set transformer: A framework for attention-based permutation-invariant neural networks,” ICML 2019.
- [3] A. Milan et al., “MOT16: A benchmark for multi-object tracking,” arXiv:1603.00831, 2016.
- [4] M. Cuturi, “Sinkhorn distances: Lightspeed computation of optimal transport,” NIPS 2013.
- [5] K. He et al., “Deep residual learning for image recognition,” CVPR 2016.
- [6] M. Zaheer et al., “Deep sets,” NIPS 2017.